

tion module 315 as described above with respect to FIG. 3. The gesture detection process 315 may utilize the data produced from any component of the system. The final coordinates 314, image coordinates 310 and 311, or a combination of 310, 311, and 314, may be sampled over time and provided as input to the gesture detection process 315. A variety of gestures (for example, “hovering” and “poking”) have been successfully detected using this data as input to a gesture detection process 315.

[0122] In scenarios where the application’s state (that is, whether or not the indicator 1101 is over a button 1109) is known and is conveyed to the gesture detection module 315. One gesture that the user performs to indicate the intention to activate the object (for example screen objects 1109, 1202) that is under the cursor 1101 is to cause the cursor to hover over the object (examples 1109, 1202) for longer than a predefined duration. This gesture performed by the user is detected by monitoring the application’s state and triggering the gesture when the application state remains unchanged for the predetermined duration. The application need not be created specifically for the multicamera control system 100, as techniques exist that can unobtrusively monitor an application’s state (in the Windows operating system by setting a “hook” using the Windows SDK function “SetWindowsHookEx”) and emulating a mouse “click” (in the Windows operating system by using the Windows SDK function “SendInput”).

[0123] In some scenarios, the application state may not be available and may not be monitored. In this case, some exemplary gestures that indicate the intention to active the object (for example screen objects 1109, 1202) under the cursor 1101 are holding the hand stationary (“hovering”), or poking the hand quickly forward and back.

[0124] A method by which “hovering” has been detected is by keeping a history of the position of the object of interest 105, where that history contains all records of the position and state for a predefined duration of time, ending with the most recent sample. That duration represents the minimum duration that the user must hold the hand stationary. The minimum and maximum position, separately in each of the three (x,y,z) dimensions, is found within the history. If the object of interest 105 was present within the region of interest 103 in all samples of the history, and the distance between the minimum and maximum is within a predefined threshold for each of the three dimensions, then the “hovering” gesture is reported. Those distance thresholds represent the maximum amount that the object of interest 105 is allowed to move, plus the maximum amount of variation (or “jitter”) expected to be introduced into the hand position by the various components of the system. The typical method in which this gesture is reported, where the system is emulating a mouse as described above, is to emulate a mouse “click.” Gestures representing additional operations of the mouse, “double clicks” and “dragging,” have also been detected and those operations have been emulated.

[0125] In addition, gestures that are independent of the position of the indicator relative to an object may optionally be detected and given meaning by the application that may or may not be dependent on the application’s state. An application that uses this style of interaction typically does not explicitly use or display the object of interest’s position 317 or other positions. These applications can be wholly or

primarily controlled with only the interpretations of the positions made by this system. These applications also need not be created specifically for this system because the interpretations made by this system can be used to simulate an action that would be performed on a traditional user input device, such as a keyboard or joystick.

[0126] Many useful interpretations depend directly on the absolute position of the object of interest 105 within the region of interest 103. (Alternately, the indicator position 1105 within the sub-region 1103 may be used in an equivalent manner). One method of making these interpretations is to define boxes, planes, or other shapes. A state is triggered on if the position (for example the position defined by block 314, or alternately by the remapped coordinates from remapping process 317) of the object of interest 105 is found to be within a first box (or beyond the border defined by the first plane), and had not been in the immediately preceding observation (either because it was elsewhere within the region of interest 103, or was not detected). This state is maintained until the hand position is not found to be within a second box (or beyond the border defined by the second plane), at which time the state is triggered off. The second box must contain the entire first box, and is typically larger. The use of a larger box reduces occurrences of the state unintentionally triggering on and off when the object of interest 105 is detected to be near the border of the boxes, where a very small motion or minor noise in the image signals would otherwise cause the position 317 to otherwise drift in and out of the box. Typically one of three methods of interpreting this state is used, depending on the intended use of the gesture. In one method, the gesture directly reflects the state with an on and off trigger. When emulating a keyboard key or joystick fire button, it is “pressed” when the state is triggered on, and “released” when the state is triggered off. In another method, the gesture is only triggered by the transition of the state from off to on. When emulating a keyboard key or joystick button, the key is “clicked.” Although the duration and off state are not reported to the application, they are maintained so that the gesture will not be repeated until after the state is triggered off, so that each instance of the gesture requires a clearly defined intent by the user. A third method is to trigger the gesture when by the transition of the state from off to on, and to periodically re-trigger the gesture at predefined intervals so long as the state remains on. This emulates that way in which, holding a key down on a keyboard, causes the character to repeat in some applications.

[0127] One way in which boxes or planes, for the above techniques, may be defined within the region of interest 103 is as follows. By defining a first plane (1501 in FIG. 13A) and second plane 1502 that divides the region of interest into “fire” 1503 and “neutral” 1504 regions (the gesture reported when the object of interest 105 is in the region 1505 between the planes depends on the previous positions of the object, as described above), the above technique can detect the object of interest 105 (typically a hand) “pushing” forward, which is one gesture for emulating a fire button on a joystick, or causing the application to respond in a way that is commonly associated with the pressing of a joystick button (for example, the firing of a weapon in a video game).

[0128] Another technique in which boxes or planes, for the above techniques, may be defined within the region of interest 103 is as follows. Planes of the first type 1506, 1507,